

УДК 004.8

Поздняков О.А.¹, Пархоменко А.В.²

¹ асп. НУ «Запорізька політехніка»

² доц. НУ «Запорізька політехніка»

МІГРАЦІЯ КОРИСТУВАЦЬКОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПРИ ІНТЕЛЕКТУАЛЬНОМУ РЕІНЖИНІРИНГУ СКЛАДНИХ КОМП'ЮТЕРНИХ СИСТЕМ

Ключовими особливостями інтелектуального реінжинірингу є застосування передових технологій, глибокий аналіз, комплексний підхід до вдосконалення та забезпечення ефективності складних комп'ютерних систем (КС). Сучасний підхід до реінжинірингу на основі передових технологій та глибокого аналізу дозволяє підвищити ефективність, гнучкість та керованість таких систем.

Як показали дослідження, при реінжинірингу КС та переході на нові версії програмного забезпечення (ПЗ) фахівці стикаються з такими проблемами:

- необхідність реінжинірингу користувацьких розробок до та після їх міграції;
- міграція даних;
- аналіз сумісності систем в ландшафті.

В роботі розглянуто існуючі та перспективні підходи при міграції користувацьких розробок (користувацького програмного коду).

При переході на нову версію КС неминуче виникає проблема з великою кількістю розробок користувача, які перекривають функціональні прогалини поточної версії КС (специфічні бізнес-процеси, вимоги законодавства, розширення стандартних функцій КС, інтерфейси). Зазвичай зі стандартним програмним кодом проблем не виникає, тому що розробник ПЗ КС відстежує зміни в моделі даних, вносить зміни в код в рамках реінжинірингу функцій КС або в результаті змін в мові програмування. В той же час, всі користувацькі розробки повинні бути проаналізовані та адаптовані співробітниками компанії або підрядником для коректної роботи в новій версії КС. Ситуація може ускладнюватися через те, що підприємство змінює свою ІТ-стратегію та переносить всю або частину КС у публічні хмарні версії КС, які часто висувають додаткові жорсткі вимоги до користувацького коду в порівнянні з локальною версією (on-premise) КС та приватною хмарою (наприклад, SAP Public cloud vs SAP Private cloud).

Найчастіше аналіз і модифікація користувацького коду відбувається в «ручному» режимі, що спричиняє використання великої кількості людських ресурсів, часу і грошей. У підсумку, терміни міграції і бюджет переходу на

нову версію КС залежать від термінів і трудовитрат на аналіз і адаптацію розробок користувачів до нової версії.

Одним з варіантів вирішення цієї проблеми є максимальна автоматизація процесів аналізу та модифікації коду.

Аналіз ринку і практичний досвід показали, що деякі компанії почали використовувати інструменти для статичного аналізу існуючого коду, що дозволяє значно скоротити час на аналіз і модифікацію користувацького коду. Деякі з цих інструментів використовують аналізатори (наприклад, Embold, SonarQube, SAP ABAP Test Cockpit), але при переході на нові версії КС, крім аналізу коду, необхідно враховувати зміни в мові програмування, моделі даних, стандартному програмному коді нової версії КС.

Прикладом є реальний кейс інтелектуального реінжинірингу систем під час переходу від SAP ERP до SAP S/4HANA при якому особливу увагу варто приділити міграції даних і користувацькому коду на мові ABAP.

Першочерговими завданнями при міграції на SAP S/4HANA в контексті цієї роботи є проведення аналізу поточного стану користувацького коду, версій мови програмування, синтаксису і структури бази даних. Найкращі практики перетворення та адаптації до нової версії коду, що реалізовані в SAP Readiness Check та ABAP Test Cockpit (ATC), вирішують завдання, що пов'язані з оцінкою поточного стану, а також виявленням фрагментів коду для адаптації.

ATC використовує дані виклику функцій для ідентифікації невикористовуваного коду, щоб він не був перенесений у нове середовище, і рекомендує виправлення коду, які потребують коригування в SAP S/4HANA. ATC дозволяє прискорити аналіз користувацького коду, зменшити витрати на аналіз та модифікацію коду, але для деяких проєктів час переходу на нову версію КС та витрати на проєкт міграції залишаються достатньо високими. Тому, для реалізації концепції інтелектуального реінжинірингу пропонується підхід на основі аналізу та модифікації користувацького коду за допомогою машинного навчання.

Методи машинного навчання, включаючи embedding-моделі для аналізу патернів і пошуку схожих фрагментів коду між старою і новою версією КС, графові нейронні мережі (GNN) для дослідження структурних залежностей між об'єктами коду, а також обробка природної мови (NLP) для обробки коментарів і документації, що поєднують статичний аналіз з методами глибокого навчання, допоможуть поліпшити якість користувацького коду при міграції складних КС.

Отже, аналіз особливостей інтелектуального реінжинірингу КС показав, що актуальною задачею є розробка та використання автоматизованих рішень на основі машинного навчання при аналізі та міграції програмних розробок користувачів, що може значно підвищити ефективність міграції та переходу

до нових версій КС, використовуючи всі переваги сучасних технологій штучного інтелекту.